Constructing a SenseCam Visual Diary as a Media Process

Hyowon Lee, Alan F. Smeaton, Noel O'Connor, Gareth Jones, Michael Blighe, Daragh Byrne,
Aiden Doherty, and Cathal Gurrin
Centre for Digital Video Processing & Adaptive Information Cluster,
Dublin City University
Ireland.
{Alan.Smeaton}@dcu.ie

Abstract

The SenseCam is a small wearable personal device which automatically captures up to 2,500 images per day. This yields a very large personal collection of images, or in a sense a large visual diary of a person's day. Intelligent techniques are necessary for effective structuring, searching and browsing of this image collection for locating important or significant events in a person's life. In this paper we identify three stages in the process of capturing and structuring SenseCam images and then displaying them to an end user to review. These stages are expressed in terms of the canonical process stages to which they correlate.

1 Introduction

Although digital cameras, camera phones and CCTV have resulted in ubiquitous photo capture as part of our daily lives, the concept of "passive capture" is still new and practical applications of this are not yet available today. Instead of a user intentionally triggering to record a moment (for example by taking a photo), a passive capture device automatically records while the user goes about her other activities, thus freeing her from having to interrupt the activity to perform the task of capturing. As the technology to develop smart sensors, smaller hardware and longer-lasting batteries improves, an advanced passive photo capture device will allow us to record our daily activities in detail while we can forget about the photo-taking activity itself and instead focus on whatever our main activity is. The photos captured in this way can then be used for reviewing, as a memory aid, or as a replacement to conventional photos for reminiscing about past events with families and friends.

The SenseCam [4] is a small wearable personal device which incorporates a digital camera and multiple sensors, including sensors to detect changes in light levels, an accelerometer to detect motion, a thermometer to detect ambient temperature, and a passive infrared sensor to detect the presence of people. These sensors detect changes in the environment and automatically trigger a photo capture at appropriate moments. The SenseCam is used to record a visual lifelog or diary of a wearer's activities. Photos are captured and stored every 50 seconds by default unless automatically triggered by the sensors or the user intentionally presses a button to capture a photo, usually resulting in up to 1,500 - 2,500 photos per day, effectively chronicling what has been happening in front of the wearer while the SenseCam was worn. Wearing the SenseCam for days, weeks, months, or even years will thus yield a detailed visual archive of a person's life.

While capturing this number of photos means the details of an event or a day have been well archived visually, an important issue arises regarding the mechanisms for the wearer to access the images later. For example, looking through 2,000 photos taken on a single day will take 10 minutes if very quickly flipped through at the speed of 0.3 seconds per photo. Searching for a particular photo will be even more difficult as the archived photo set increases.

The SenseCam image management system we have developed is a prototype application for SenseCam usage that resolves some of the problems of managing SenseCam images. The system employs a number of content-based image analysis techniques to automatically structure and index the captured photos in such a way that the owner of the photos can easily search and browse the large amount of SenseCam photos through a web-based interface.

Although there is no similar system in use anywhere else, we have built a robust end-to-end system in order to investigate issues that will arise from such an application in the near future. Though having no existing system in use, or a user base from which details of user needs and requirements could be obtained, our approach has been to build the scenario and the physical application, and then get a small number of early adopters to actually use it for a period of time so that we can refine the system based on that usage [5].

In this article we try to understand our SenseCam system as a media production and utilization tool, in particular in terms of the Canonical Media Process. By aligning our system's processes with the Canonical processes, we can more fully understand the process cycles of our system in the context of the more generalised, standard process cycles of existing systems, and thus we can envisage future scenarios where some of the processes within our system could be exchanged with those from other media production systems.

In the next section, we describe the use of our SenseCam image management system, how it automatically processes the captured photos, and how the user can subsequently access them. Then in Section 3 we describe the SenseCam system in terms of the Canonical processes and we try to match the individual processes. In Section 4 we discuss issues arising from aligning the processes, and Section 5 concludes the article.

2 Chronicling a Day with SenseCam

Using our SenseCam image management system starts with the user wearing the SenseCam device for particular event(s) or throughout day(s) and then uploading the captured photos to where automatic content-based analysis commences and subsequently allows the user to search and browse them on a web interface. In this section, we describe this procedure in more detail.

2.1 Capture and Upload

The user wears the SenseCam for a period of time during which it will continuously capture and store photos. At the end of the event or the day, the user connects the device to a PC from which all captured photos are uploaded.

2.2 Processing

Once captured photos have been uploaded to a PC for processing, a series of automatic, content-based image analysis techniques are applied to the photos in order to structure them into an accessible index. The three main processing elements are Event Segmentation, Landmark Photo Selection, and Calculation of Novelty Values for Events.

Event Segmentation
The first processing step involves the segmentation of all photos into distinct groups, or events, e.g. having breakfast, talking to a work colleague, meeting a friend at a restaurant, etc. To achieve this goal, *context-based* sensor analysis is used in conjunction with *content-based* image analysis.

SenseCam photos are taken at quite a low frequency (one image every 50 seconds) in comparison to video data (20 images per second), and naturally there can be quite a degree of visual change between successive images in the same event (e.g. while sitting at her PC, the SenseCam wearer may turn away from her screen for a short break). Using traditional shot boundary techniques from the video domain, "outlier" images in an event would in fact indicate a new event if only adjacent images are compared. To segment a group of images into events using content-based image analysis, an adaptation of Hearst's Text Tiling approach is used [3]. Using this technique, for a reference image, the block of images previous to it is compared to the block of images following it; each block represents the average values of the low level MPEG-7 visual features (colour structure, colour layout, scalable colour and edge histogram) for all the images in that block. By taking the average value of images, the difficulties in dealing with high variability among images within an event and effect of outlier images is reduced.

Photos can also be segmented into events by using context-based sensor analysis. In essence the derivative values of sensors can indicate the likelihood that the wearer may have have changed activities, e.g. a change in motion values is likely to occur when the wearer has been sitting down at work but then starts walking to go to lunch. Similarly there may be a change in the ambient temperature when the wearer moves from outdoors to indoors, and there may also be significant changes in light levels too. The SenseCam image management system segments images into events based on fusing the output of content-based image analysis with the output of corresponding context-based sensor analysis [2].

Landmark Photo Selection
The second step in processing SenseCam images involves the selection of a landmark photo for each event, namely a single photo from within an event which represents the event's content. Two approaches were investigated with the first being the selection of the image whose visual features most closely resemble the average value across the entire set of images in the reference event. For the second approach the middle image from the event is selected as the landmark photo. In practice we found little difference between both approaches and currently in our system we implement the latter approach, though this is a topic for further investigation. For example, an image that contains a large or well-focused face could be selected as the landmark image within an event, using an automatic face detection algorithm; selecting the images that have medium level of brightness (not too bright and not too dark), or selecting less blurry ones will also reduce the choice of landmark image selection that will help the users review their photos.

Calculating Event Novelty
The final processing step calculates how important, or more accurately how visually unique, each event is. The concept behind this is that when reviewing the past events or days, the user will derive greater benefit in seeing more unusual or unique events (meeting an old friend, a picnic, a presentation at a conference, etc.) than commonly occurring events (sitting at the desk working, walking in the corridor, bus trip to and from work, etc.).

As in the previously mentioned landmark photo selection processing stage, each event is represented by the average bin values of the low level MPEG-7 visual features across all the images in that block. Event representative vectors are compared using the Manhattan distance metric to determine the similarity between any two given events.

To determine how unique an event is, a form of novelty detection is used. Essentially the most dissimilar event in a given period of time, in comparison to other events in the same period, is selected as the most unique event in that period. The system described in this article calculates uniqueness values as follows: each event in a week is compared to see how dissimilar it is to every other event in that week. The event that is most dissimilar to the other events in the week is judged as the most unique event. For the day in question the event uniqueness scores are normalised against the most unique event in that day.

Following these three processing steps the system is then ready for the user (the owner of the SenseCam images) to access the photos to review, browse and search their past activities.

2.3 Accessing the SenseCam Photos

In order to achieve positive benefits from archiving a large number of photos in this way, we need to have ways to access the accumulating photos. Our online SenseCam image management system automatically structures hundreds of thousands of SenseCam photos so that the wearer can browse, search, annotate or save for future reference. This means that the user can have a concise overview of any day's events presented on a single page (see the middle column of Figure 1). A small number of significant or unique events for the day are automatically selected and a landmark image from each event is presented in different sizes depending on the novelty value of the event. The system uses a packing algorithm similar to [1] to automatically compose a layout for the photos with five different sizes in a compact manner. The number of events shown is 20 by default, but the user can drag the slider bar (see near the top of middle column in Figure 1) to adjust this. A larger photo (higher novelty value) naturally gets more attention from the user, whereas a smaller photo (lower novelty value) is more easily neglected. A mini calendar widget is provided (see the top-left of Figure 1) from which the user can select any particular day, week, month, or arbitrary number of dates. The automatically selected events will be then displayed on the middle column summarising whatever number of dates are selected.

The user can move the mouse cursor over an event to see all photos

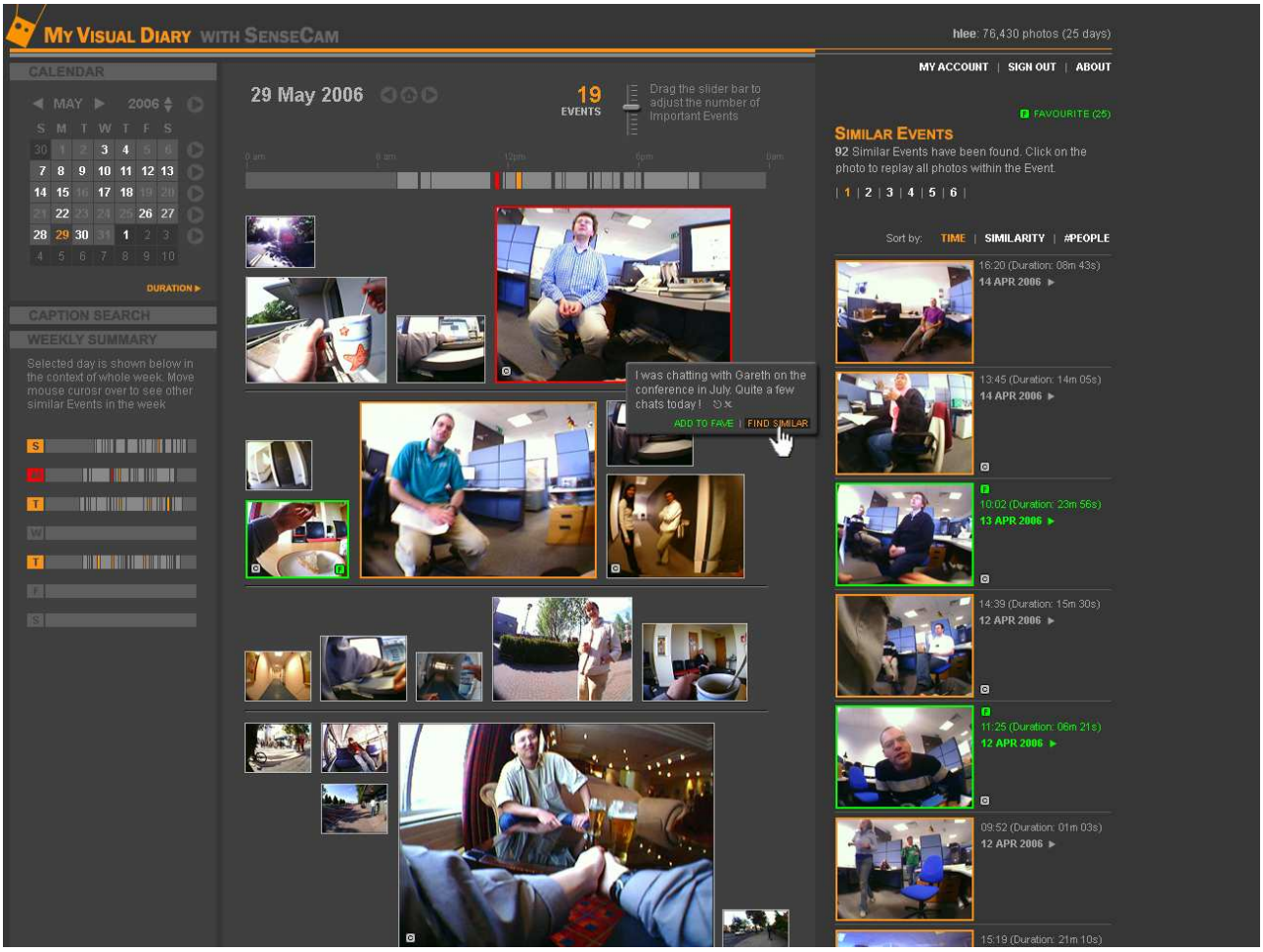
MY VISUAL DIARY WITH SENSECAM
hlee: 76,430 photos (25 days)
MY ACCOUNT | SIGN OUT | ABOUT
CALENDAR
29 May 2006
19
EVENTS
Drag the slider bar to adjust the number of important Events
SIMILAR EVENTS
92 Similar Events have been found. Click on the photo to replay all photos within the Event.
Sort by: TIME | SIMILARITY | #PEOPLE
I was chatting with Gareth on the conference in July. Quite a few chats today !
ADD TO FAVE | FIND SIMILAR
CAPTION SEARCH
WEEKLY SUMMARY
Selected day is shown below in the context of whole week. Move mouse cursor over to see other similar Events in the week
DURATION

Figure 1. Interactive SenseCam Photo Browser

within that event slideshowing at a user-customised speed (default is 10 photos per second). The timeline above the group of photos orientates the user by highlighting in red the time of the day of the event where the user has currently placed the mouse cursor, while at the same time highlighting in orange other similar events that happened on that day. The user can also specify an overview of multiple days by selecting a number of dates from the calendar (see top-left of Figure 1).

As well as slideshowing all photos within an event, a pop-up menu is presented with a few useful menu options. If she had annotated the event before, the annotation text appears on this menu which can be edited or removed (see the red-framed image in the middle of Figure 1). The user can also set an event as a "Favourite" event, similar to the "bookmark" feature of a web browser, so that it can be easily accessed later. The "Find Similar" menu option on this pop-up menu can be selected to retrieve all similar events to the selected one, to be presented on the right column of the screen. In Figure 1, the user is looking at the event where she is chatting with a colleague in the lab and has clicked on the "Find Similar" button on the menu to see all other similar events to chatting with that colleague, which are then presented on the right column.

The user can annotate in text each event, or edit, or delete if so wished. Annotated text can be searched by clicking and opening the 'Caption Search' panel just below the calendar.

The initial live version of the system has been available to a limited number of people within our group and outside who have been wearing the SenseCam for some period of time (ranging from 1 month to 15 months), and they have been using the SenseCam image management system to get their photos automatically indexed and to use the web-based interface for browsing their daily photos. For example, the system is being used to help an ethnographic study being conducted in Tampere, Finland, in capturing the details of biomedical practitioners' information seeking behaviour throughout the day, by having some of the practitioners wear the SenseCam during their work. Although still at early stage for a full deployment to wider users due to the limited availability of SenseCam, our initial users will continue using the system during which we will monitor their usage.

3 Relationship with Canonical Media Process

From the capturing of the photos when wearing the SenseCam, to the access via a web interface, the way the medium (large number of photos taken by SenseCam) is captured, used and consumed corresponds to Canonical Media Production Processes. In this section, we describe our SenseCam image management system in terms of the Canonical Processes.

Overall the Capture and Upload (Section 2.1) corresponds to **create**, **annotate**, Processing (Section 2.2) corresponds to **annotate**, **construct message**, **package**, **organise**, and Accessing (Section 2.3) corresponds to **organise**, **construct message**, **publish** and **distribute**. Some processes can be carried out automatically by the system and/or interactively by human, and the time that these happen can be varied. For example, **organise** initially happens by the system but later can happen at interaction time and the end-user can influence the way the information is organised and presented in our system, whereas the **annotate** process is sometimes automatic (by the system, before and during user interaction) and sometimes manual (by the user, during the interaction). Some of these issues will be dealt with in Section 4.

3.1 Canonical Processes for Capture/Upload

The user wears the SenseCam device because she conceives its value as a visual archive tool for recording what happened during a particular event or throughout her daily life (**premeditate**). The current implementation of the device requires a certain degree of determination or commitment from the user, as the wearing of the device throughout the day can be inconvenient and daily uploading of the photos to the system can be a chore. Smaller hardware that could be more comfortably worn, for example as a brooch or a button, and a wireless automatic daily uploading of the photos to the web server, would both enhance its usability. Alternatively, the user may decide to wear the device only on special occasions such as a tourist trip, holiday or a wedding. If an individual wears the SenseCam device in either case, it is quite likely that they wish to recall significant and memorable photos from those events. The input to the **premeditate** process is the user's wish/motivation to record images for a period of time. The output is the user's decision to wear the SenseCam and use the system, and an initial user profile that is recorded to the system once she registers. In a more personalised service, a user's motive for wearing the device and her various usage preferences could be explicitly recorded when the user decides to embark on using the system (for example at the time of online registration to the system), and this becomes the output of the process to be used for presenting an appropriate browsing interface suitable for the particular task for which a particular motive of the user has incurred.

Wearing the SenseCam throughout the day automatically generates a number of photos (**create**), stored on the device. In addition to the capture of JPEG images, other sensed data (such as ambient temperature and light levels) is recorded on the device, to be associated with each photo along with time stamps, creating the initial set of metadata associated with the photos (**annotate**). A more central role of the **annotate** process occurs during the processing time and the user interaction time. At this stage the input to the **create** process is the sensor data itself which gets stored on the device, and the output is the captured photos and the sensor data.

3.2 Canonical Processes for Processing and Indexing

Photos with their initial metadata go through a series of content-based analyses to add further machine-generated descriptions as described in Section 2; similarities among the photos are analysed to determine the boundaries of individual events; a landmark photo is then determined from each event; and the novelty value is calculated for each event by comparing the similarity between all events on that day and all other events that occurred during the previous week. From this, the associations among events are established, both within a day and within the window of the past one week period. These processes result in additional metadata (**annotate**) and using this additional metadata the system establishes the concrete grouping of photos into events so that these can be retrieved as the main units of searching and browsing (**package**). Most of this process is logical packaging as the generated metadata are separate from the actual stored photos and are only marked-up information that point to different parts of the photo set.

These are automatic processes that add partial descriptions of the photos in order to kick-start the initial organisation and presentation of photos to the user, and are thus purely *machine-generated annotations*. Subsequently, metadata may be further enriched with *human-created descriptions* as a user provides further metadata during interactive searching and browsing at a later stage.

The input to the **annotate** process is the images with sensor data and the output is the time-stamped images with annotation which becomes the input to the **package** process, in turn resulting in the images grouped into distinct events as output.

Currently, discussing the input and output between the **annotate** and the **package** processes is not an important issue, as these two processes are tightly bound together and coded as one module in the system. However, when alternative packaging methods become available (for example, using different algorithms for better event segmentation, or grouping of the day's photos into different conceptual units than event) by ourselves or from different multimedia indexing systems, it will be possible to use identical output from the **annotate** process and use an alternative functional module for the **package** process in order to generate the images grouped in different ways.

3.3 Canonical Processes for Accessing the Photos

The user visits the online SenseCam image management system to review past events or past days in general. She can query by selecting a particular date, or by typing in text terms to match against annotation text she had added previously (**query**). The result is a structured set of pointers to the subset of the archived photos. For example, when the user selects a particular date or a number of dates from the calendar, the packaged events that fall into the specified period are selected and ordered (**organise**), then the information associated with those selected events such as user annotation, "Favourite" events and user account settings are extracted to refine the selected events for final presentation (**publish**). In the case of the query-by-example style query, called "Find Similar" in the system, the user selects one event that initiates retrieval of similar events (**query**). The result is a ranked list of pointers to groups of photos (**organise**), ranked by the degree of match based on content-based similarity measures processed during the **package** process. The input to the **query** process is either a specific date range, or text query terms, or an example image that represents an event. The output is a list of events that fall into the specified date range, or those events whose user annotation text matches the query terms, or those events that are most similar to a given query event.

Those events which are a result of the user's query are then organised according to the system- or user-set presentation parameters (described later), for the final presentation to the user (**organise**). This involves creating an interface template that specifies the sizes of each image according to the novelty values assigned to each of

the selected events, the association of other attributes to events (e.g. duration, "Favourite" events, their text annotation, etc.) in order to prepare for presentation. The input to the **organise** process is the list of events with the novelty values. The output is the visual summary of landmark events to be displayed to the user.

In the **publish** process, the organised internal information is then displayed on the web interface by collecting necessary metadata such as the type of each event, "Favourite" icons, the novelty value for each event, a comic-book style layout dynamically generated by a packing algorithm, a particular CSS (Cascading Style Sheet) to be used for the session, and other presentation parameters (the number of events to be displayed on a screen, the speed of image slideshow when an event is selected by the user, etc.).

Presentation parameters are set by the system designer but some of these can be changed by the user during interaction. For example, by default, the number of events presented on a page (whether it is for a single day or multiple days) is set as 20, but during browsing the user can adjust this value (**construct message**). The changed value is remembered and used for subsequent presentations for this user. As another example, the speed of slide show when the user moves the mouse cursor over an event is set to 10 photos per second by default, but we feature this as customisable by allowing the user to modify the speed of slide show. There can be many more customisable interface elements which the initial presentation uses as pre-set default values, and the user can modify these values to cater for her requirements or preferences afterwards. The input to the **construct message** process is the user's wish to modify the presentation parameters after having interacted with the presentation set by the default parameters. The output is the modified presentation settings.

Currently the system allows only the online display of query results on the desktop computer's web browser, but we are planning to add features whereby the user can save the query result screen to a PDF document or save the specified events to a DVD, as an alternative **publish** process. A similar layout and style as in Figure 1 will be used for PDF and DVD format with the main difference being the PDF and DVD presentations will have a closed subset of the available dates from the full dataset available on the web interface. However, we are aware that there will be more suitable presentation and interaction schemes for different presentation devices and interaction modalities. For example, PDF format will be more likely used for print-out on paper thus having brighter background colour with dark text, fitting a day's photos on a printable page area will be more suitable; DVD presentation, once generated, will likely be used on a DVD player connected to a TV, thus an alternative interaction style will be required for more effective user interaction when users try to use a few buttons on their remote controller while sitting back on a sofa at a distance in order to interact with such a presentation. We envisage that such device-specific, non-desktop presentations can be best accommodated by other media presentation systems that have expertise in their **publish** process for the print medium or the interactive TV platform, by taking our SenseCam image management system's **organise** output.

The possible **distribute** processes are where the printable PDF file or burned DVD is produced and actually consumed by users at home in their final format (e.g. sharing of printed papers among family members, or by loading the DVD on their DVD player and watching it on a TV screen).

The user can add text annotations for each event to further add value to her photos and to help future searching and browsing (**annotate**), thus in turn propagating back to the **package** process. Tagging an event as "Favourite" is also a way of adding annotations for future retrieval (**annotate**), influencing subsequent **organise** and **publish** processes.

4 Discussion

While mapping our SenseCam application onto the Canonical Media Process, we identified a few important issues that were raised. These are discussed in this section.

Interactivity
Some of the Canonical processes can happen either off-line or during user interaction. For example, the **organise** process happens initially as soon as the photos are uploaded to the system, but as the user queries or adds more annotations, the **organise** process is triggered again to generate specific ordering and structure for the resultant set of photos to be presented to the user. As another example, the **annotate** process in our system starts with the system's automatic enrichment of metadata onto the photos but once the user starts interacting with the system, she can further enrich the metadata by manually adding captions to events, which in turn will influence subsequent **organise**, **publish**, and **distribute** processes.

In general, the SenseCam image management system's processing cycle can be divided into the following two phases:

1. Phase I: Initial offline processing, machine-processed
2. Phase II: Interactive augmentation of information during a user's browsing and searching

The Canonical processes **create**, **package** and **organise** are triggered at Phase I, initially automatically by the system. Once the user starts interacting with the system (Phase II), the processes **query**, **organise**, **publish** and **distribute** are triggered almost as a single process.

One of the reasons why the mapping between the use of our SenseCam image management system and the Canonical processes may not seem straightforward is due to user-system interactivity where automatic photo annotation and re-organisation can happen during the interaction.

Automatic Processes vs. Manual Processes
The Canonical process framework does not specify how each process is executed, thus allowing manual processing, which is gradually changed to automatic processing as technology advances in the future. In the context of our SenseCam image management system, however, our premise is to regard automatic processing as the main strength of the developed system (automatically organising the large number of photos to reduce the user's browsing burden), and in case the human user wants to refine results manually, the system should allow this. While the current version of the interface does not feature this, we plan to add features where the user can correct the results of the automatic processes where necessary. For example, if the user notices two sub-events while browsing an event, this can be perceived as incorrect machine processing and thus the user could correct this by manually segmenting the event into two. There will be cases where the machine's automatic processing is not necessarily to blame arising from the users' subjective view to what constitutes an event. Perhaps personalisation techniques could be incorporated into the automatic processing of SenseCam images so that the meaning of "accurate automatic processing" is one that

is capable of interpreting a user's individual views.

On the other hand, if we think a perfect machine-processed annotation process will reduce the user's manual annotation effort, there will be a gradual shift of importance from the user's manual annotation to the system's automatic annotation, as the overall aim of our SenseCam image management envisages, and the Canonical framework accommodates.

System Designer's Intention vs. User's Intention
In any technical system that processes media, there is inevitably a large number of system parameters that influence the organisation, retrieval performance and the presentation to users. In the case of the SenseCam image management system, the system designer needs to make decisions during the system development stage on various system parameters such as the threshold values for determining event boundaries, the weighting scheme for combining visual evidences in calculating overall photo-photo similarity, and the decision on the amount of past data required to determine the visual uniqueness of a particular event. These can be referred to as the system designer's intentions, and are incorporated in the **`package`**, **`organise`**, and **`publish`** processes of the Canonical process.

In an existing system that already has a concrete user base, one would expect those system parameters that affect end-users to be set based on empirical user studies in such a way that will support users and their current practices. The SenseCam image management system is a new system that has not been used by many users yet, thus empirical evidence for setting parameters is not possible. For those parameters that directly influence user viewing and for which there is no proven optimal setting, user-customisable parameters can help, as usually prescribed by the area of Human-Computer Interaction [6]. For example, during the **`construct message`** process the user can modify the number of events to be displayed on-screen, or the speed of the slideshow when she places the mouse cursor over an event, explicitly conveying her intention to the system in order to improve the current presentation specifics. Similarly, we can imagine a separate customisation panel where the user can indicate her intentions for many different parameters, in effect making the **`construct message`** process a feature whereby the parameters set by the system designer's initial intentions (by default) can be replaced by the user.

More Complete Provision of Features
Systems such as our SenseCam image management tool, have little precedence in their development or usage, thus we cannot be informed by the existing practise or experienced designers. Many of the user-oriented, useful features for satisfying end-user experiences are probably missing. Having a more generalised, complete chain of media processes provided by the Canonical process, we can identify the elements in the developed system that probably need further elaboration or further provision of features, making the overall system provision more complete.

Technically speaking some of the SenseCam system processes do not have clear boundaries between each other. For example, as we have mostly focused on an interactive web-based presentation in our development of the system, the **`organise`**, **`publish`** and **`distribute`** processes have been coded as one technical process with no particular function for **`distribute`** process.

Thinking about the **`publish`** and **`distribute`** stages as a separate process after the **`organise`** process, we are forced to think about other possible ways of presentation which we had not thought about before. For example, after the **`organise`** process (as a result of user's query), could the result be presented via mobile phone or email? Can we have a button on the web interface to print out a comic-book page of the day, or burn a DVD of a month's visual diaries to take away or just to have as physical objects, or a Print a year book? The aforementioned plan for saving a visual summary in PDF format or requesting a physical DVD version was triggered by trying to separate our "present query result" into multiple processes, as the Canonical Process suggests.

By formally thinking about the **`publish`** and **`distribute`** processes, their possible alternative functions, and the distinction between the two processes, more elaborate new features such as these arose complementing the relatively weak **`publish`** and **`distribute`** processes of our initial system design.

Once these are more clarified, the output of pre-presentation processes could be plugged into *other media production systems* that have more specialised **`publish`** and **`distribute`** processes, as mentioned in the previous section. One of the advantages of implementing the Canonical Media Process is thus being able to leverage the specialty of different media production systems, but in developing the novel SenseCam image management system we benefited by being able to come up with more complete features that will add value to the system when used in a real context, especially how our system's **`publish`** and **`distribute`** processes could be more expanded or adopted from other systems for different modalities and interaction platforms. Similarly, photos or home video type of personal media data from other people could be "pumped" into our **`organise`** process to benefit from our system's automatic organisation and presentation methods, provided that those external media has same input structure as enforced by the Canonical Processes.

5 Conclusion

In this article, we have examined how a novel system for management of SenseCam images can be understood in terms of Canonical Media Processes. We have identified different stages of SenseCam image use which currently assume little contribution but could be expanded further, such as **`publish`** and **`distribute`**. These are potentially the processes that the output from our SenseCam image management system could be plugged into, and combined with some other system's processes in order to generate appropriate presentation for the end-users.

Also we envisage that an **`organise`** process which has more advanced metadata generation schemes and perhaps true semantic annotation (as opposed to the current text strings as users typed in their annotation) could be plugged in to the system in near future, in order to provide the users with more semantically meaningful searching and browsing features.

ACKNOWLEDGEMENTS

The SenseCam project at Dublin City University is supported by Microsoft Research and by Science Foundation Ireland under grant No. 03/IN.3/I361.

6 References

[1] J. Boreczky, A. Girgensohn, G. Golovchinsky, and S. Uchihashi. An interactive comic book presentation for exploring video. In *CHI '00: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 185–192, The Hague, The Netherlands, 2000. ACM Press.

Canonical Process	SenseCam image management system
Premeditate (1)	The user deciding to use a wearable camera to capture images of a day's events or of significant events such as going to a wedding, birthday party, zoo which (s)he would like to be recalled and reviewed at some later stage. **Input**: user intention/motivation to record images. **Output**: decision to wear the device and access the system; initial user profile that the user provides on the initial online registration form.
Create Media Asset (2)	Images passively captured by the SenseCam wearable camera. **Input**: sensor data that triggers automatic photo capture. **Output**: raw images along with sensor file
Annotate (3)	All images automatically time-stamped on download of images from SenseCam to computer. **Input**: output of (2). **Output**: time-stamped SenseCam images. Automatically describe each image in terms of its low-level visual features such as colour, texture, or edge. **Input**: time-stamped images from computer. **Output**: annotated images. During user interaction, user adds text annotation to events to add value to his/her archive.
Package (4)	Automatically segment the annotated images into events. **Input**: annotated images from (3). **Output**: images organised into events.
Query (5)	User selects a particular date, week, month, or a range of dates; or types in text query; or requests all similar events by clicking on Find Similar button. **Input**: images from (4) and query statement (specific date range or text query terms or an example image representing an event). **Output**: a list of events and their relative uniqueness rating.
Construct Message (6)	Setting display options such as the number of events to be displayed on the page; setting the speed of slideshow on each event, etc. **Input**: user's intention to modify the presentation parameters to suit her interaction/viewing style and preferences. **Output**: modified set of presentation parameters, to be used for this user once the process occured.
Organise (7)	Creating an interface template that will emphasise the most important events in a visual manner to the user. **Input**: list of events with importance values from (5). **Output**: summary of landmark events to be prepared for display.
Publish (8)	Selecting appropriate events and their metadata to be presented on the screen (web interface). Alternatively, generating appropriate formats in PDF file (for print-out) or in DVD (for offline interactive browsing on TV screen); generating downsized versions suitable for a mobile phone or PDA consumption. **Input**: results of process (7) and all associated presentation-specific information. **Output**: generated presentation formats.
Distribute (9)	Displaying the interactive visual summary on the web interface; printing out the daily summary in PDF format; inserting the generated DVD on interactive TV and browsing with a remote controller; transmitting to a mobile phone or a PDA, etc. **Input**: results of process (8). **Output**: viewing on a web browser, on a printed paper, or interacting with a DVD player.

Table 1. Description of dependencies between visual diary stages and the canonical process for media production

[2] A. R. Doherty, A. F. Smeaton, K. Lee, and D. P. Ellis. Multimodal segmentation of lifelog data. In *RIAO 2007 - Large-Scale Semantic Access to Content (Text, Image, Video and Sound)*, 2007.

[3] M. Hearst and C. Plaunt. Subtopic structuring for full-length document access. In *SIGIR - The 16th Annual ACM-SIGIR Conference on Research and Development in Information Retrieval*, 1993.

[4] S. Hodges, L. Williams, E. Berry, S. Izadi, J. Srinivasan, A. Butler, G. Smyth, N. Kapur, and K. Wood. Sensecam : A retrospective memory aid. In *UbiComp 8th International Conference on Ubiquitous Computing*, 2006.

[5] H. Lee, A. F. Smeaton, N. O'Connor, and B. Smyth. User evaluation of Fischlar-News: An automatic broadcast news delivery system. *TOIS - ACM Transactions on Information Systems*, 24(2):145–189, 2006.

[6] B. Shneiderman and C. Plaisant. *Designing the User Interface: Strategies for Effective Human-Computer Interaction (4th Edition)*. Pearson Addison-Wesley, Reading, MA, 2004.